

Motion Imagery Standards Board Standard 0301.5: MISB Profile for Aerial Surveillance and Photogrammetry Applications (ASPA)	MISB Standard 0301.5 Version 1.5 26 July 2010
--	---

1 Scope

This Standard documents the DoD/IC/NSGI Motion Imagery Standards Board (MISB) standard profile for the Advanced Authoring Format (AAF) and the Material eXchange Format (MXF). The profile constrains the contents of AAF and MXF files to those in accordance with the Motion Imagery Standards Profile (MISP).

The purpose of this document is to state DoD/IC/NSGI participant's requirements for AAF and MXF files to address specific operational needs and to form the basis for development of an AAF Implementation Guideline (IG) submitted to the AAF Association for approval as part of the AAF standard suite.

This version 1.5 includes support for MISB Standard 0601.4 formatted metadata and additional security metadata specified in MISB Standard 0102.8.

Statement of current practice and intent: When metadata that ASPA implementations can parse is present in an instance of contained essence, ASPA implementations shall populate faithfully the associated ASPA metadata element.

2 References

SMPTE 336M-2001, *Data Encoding Protocol Using Key-Length-Value*

SMPTE 335M-2001, *Metadata Dictionary Structure*

SMPTE RP210.12-2010, *Metadata Dictionary*

SMPTE 377-1-2009, *Material Exchange Format (MXF) File Format Specification (Standard)*.

SMPTE RP217-2002, *Nonsynchronized Mapping of KLV Packets Into MPEG-2 Systems Streams*

SMPTE 395M-2003, *Groups Registry Structure*

SMPTE RP-2009 *Groups Registry (DRAFT)*

SMPTE 400M-2003 *Labels Registry Structure*

SMPTE RP224.7-2004 *Labels Registry*

MISB *Motion Imagery Standards Profile (MISP)*, V 5.5

NGA D&R IDM Rev E-2004

NGA D&R IDM D&R IDM Revision R, 31 March 2010

MISB RP 0101, *Use of MPEG-2 System Streams in Motion Imagery*, 28 February 2001

MISB RP 0102.8, *Security Metadata Sets for Digital Motion Imagery*, 20 May 2010

MISB RP 0103.1, *Timing Reconciliation Metadata Set for Digital Motion Imagery*, 11 October 2001

MISB EG 0104.5, *Predator Metadata Sets*

MISB Standard 0601.4, *UAS Datalink Local Metadata Set*, 4 March 2010

MISB EG 0602, *MISB Metadata Registry and Processes*, (Draft)

MISB RP 0107, *Bit and Byte Order for Metadata in Motion Imagery Files and Streams*, 11 October, 2001

MISB RP 0608.1, *Motion Imagery Identifier (MIID)*, 13 December 2007

ITU-T Rec H.222 | ISO/IEC 13818-1:2000 / Amendment 1: *Carriage of metadata over ITU-T Rec H.222.0 / ISO/IEC 13818-1 streams*, March 2003 (DRAFT)

AAF Association, *AAF Specification V1.1*, November 2004

AAF Association, *AAF Specification V1.0.1*, December 2003 (bibliographic only: V1.1 is normative)

SMPTE 381M-2004, *Mapping MPEG Essence Data to the MXF Generic Container*

SMPTE EG42-2004, *MXF Descriptive Metadata*

MIL-STD-2500C, *National Imagery Transmission Format (NITF) Version 2.1* STDI-0001, *National Support Data Extensions (SDE) for the National Imagery Transmission Format (NITF)*, Version 1.3, 2 October 1998

STDI-0002, *Compendium of Controlled Extensions (CE) for the National Imagery Transmission Format (NITF) Version 3.0*, 1 August 2007

STDI-0005, *Implementation Practices of NITFS (IPON) Version 1.0*, 1 August 2007

3 Introduction

The Advanced Authoring Format (AAF) is a multimedia file format developed to promote file-level interoperability across different platforms in the digital cinema and television industry. While AAF was designed initially for the entertainment industry, the parallels between their digital production, post-production, archiving, and product distribution processes using AAF and those needed for digital motion imagery in the DoD and Intelligence Community are remarkable.

The Material eXchange Format (MXF) (SMPTE 377M) is a multimedia file format for the exchange of program material between file servers, but it is also for tape streamers and digital archives. It usually contains one complete program, but this may comprise a sequence of clips and program segments. The ‘body’ is a stream-based multimedia container, which contains a sequence of frames where each frame comprises audio, video and data essence plus frame-based metadata. AAF and MXF are interoperable because they share the same object model and the same method of defining essence

The proposed AAF Profile for Aerial Surveillance and Photogrammetry Applications (ASPA) forms the basis for development of a prototype demonstration of the AAF format in the NGA Image Product Library (IPL). Following the prototype “AAF-in-IPL” demonstration, the AAF ASPA Profile can be used by the DoD/IC/NSGI community to specify standardized commercial off-the-shelf (SCOTS) products for motion imagery processing, exploitation, archive, and distribution functions.

This document must be read in conjunction with the AAF Specification V1.1, since it does not repeat any of the contents of the AAF Specification – this document states constraints on the AAF Specification and defines necessary AAF Extensions which collectively form the ASPA Profile for AAF.

ASPA files may be stored in accordance with the MXF format specification, SMPTE 377M. All data and constraints specified herein apply equally to ASPA-MXF files as to ASPA-AAF files.

This document supports the ASPA Browser version 5.0 that uses AAF SDK version 1.1.4 and mxflib version 1.2.

4 Terminology

4.1 Notation

In this document, all terms which refer specifically to defined items within the AAF Specification and Software Development Kit (SDK) are in `Courier` font. Some examples may be seen in paragraph 4.3 below.

4.2 File Kinds

In this document, some shorthand phrases are used to avoid repetitive language:

“**File**” – means any AAF/MXF File, whether conforming to the ASPA Profile or not.

“**ASPA File**” – means an AAF/MXF File which conforms to the ASPA Profile.

“**non-ASPA File**” – means an AAF/MXF File which does not conform to the ASPA Profile.

“**Other File**” – means any file which is not an AAF/MXF File.

4.3 Manner of Specification

This document must be read in conjunction with the AAF Specification. The major sections and subsections of this document are matched with those of the AAF Specification.

This document does not repeat any of the contents of the AAF Specification. Instead, it defines three kinds of variations on the AAF specification: numerical constraints, semantic constraints and extensions.

4.3.1 Numerical Constraints

Numerical Constraints on the AAF specification limit the capacity of a File. They may be constraints on the number of a given `Object` allowed in an ASPA File, or specific ranges for given `Property` values in an ASPA File. Numerical Constraints are given in the form of tables.

4.3.2 Semantic Constraints

Each set of numerical constraints is followed by a set of semantic constraints, which serve two purposes: they give a prose explanation of the given numerical constraints, and they define additional restrictions upon combinations of `Objects` and `Property` values which are not possible to clearly tabulate.

Numerical and Semantic constraints are presented in the same order as the AAF Specification to which they relate and with corresponding major section and sub-section numbering.

4.3.3 Extensions

Extensions are specifications for `Classes`, `Objects`, `Properties`, `Types` and `Definitions` peculiar to ASPA Files that are not built in to the standard AAF Specification or SDK. All Extensions are created using the standard AAF extension model (thus, all SCOTS AAF implementations will have the capacity to process ASPA Files).

Extensions are presented after the Constraints in each major section of this document.

5 Class Packages

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

6 Structural Metadata Classes

6.1 Header

The ASPA Profile alters this AAF Class Specification as follows:

6.1.1 Numerical Constraints

The ASPA Profile does not change any numerical constraints on this class. Thus:

ASPA Files shall contain one and only one Header object.

6.1.2 Semantic Constraints

ASPA Files shall contain only the `Preface` subclass of `Header`, as defined by MXF.

6.1.3 Extensions

`Preface` defines three required properties: `OperationalPattern`, `EssenceContainers` and `DMSchemes`.

6.2 Identification

The ASPA Profile does not alter this AAF Class Specification in any way.

6.3 Dictionary

The ASPA Profile does not alter this AAF Class Specification in any way.

6.4 ContentStorage

The ASPA Profile does not alter this AAF Class Specification in any way.

6.5 Mob

The ASPA Profile alters this AAF Class Specification as follows:

ASPA Files shall contain only the following subclasses of `Mob`

`MasterMob`

`SourceMob`

Each `Mob` in an ASPA File shall contain the following numbers of Slots:

0 or 1 `TimelineSlot`, with a `DataDefinition` equal to `Picture`.

0 or 1 `StaticSlot`, with a `DataDefinition` equal to `Picture`.

1 or more `StaticSlot`, with a `DataDefinition` equal to `DescriptiveMetadata`.

– plus any other Slots specified for subclasses of `Mob` (see below).

6.6 CompositionMob

The ASPA Profile does not alter this AAF Class Specification in any way.

However, note that `CompositionMob` objects are not present in ASPA Files

6.7 MasterMob

The ASPA Profile alters this AAF Class Specification as follows.

6.7.1 Numerical Constraints

ASPA Files shall contain one and only one `MasterMob`.

6.7.2 Semantic Constraints

The `MasterMob` shall contain at least a `StaticSlot` with `DataDefinition` equal to `DescriptiveMetadata`; containing Level 0 Metadata.

Level 0 Metadata is carried in a `DMSegment` containing an `ASPA_Framework` (see 17.1 below), which in turn contains a `DM_Set_File` (see 17.3 below). .

6.8 *SourceMob*

The ASPA Profile alters this AAF Class Specification as follows.

6.8.1 Numerical Constraints

ASPA Files shall contain one top-level `SourceMob` for each `EssenceData` object in the file (a top-level `SourceMob` is one that is directly referenced by a `MasterMob`) .

ASPA Files may also contain a lower-level `SourceMob` for each `EssenceData` object in the file (a lower-level `SourceMob` is one that is referenced by another `SourceMob`) .

ASPA Files may contain additional top-level `SourceMobs` for which there is no `EssenceData` object in the file. These `SourceMobs` describe external essence. ASPA Files shall contain one lower-level `SourceMob` for each external essence `SourceMob` object in the file.

6.8.2 Semantic Constraints

Each top-level `SourceMob` shall contain at least a `StaticSlot` with `DataDefinition` equal to `DescriptiveMetadata`; containing Level 1 Metadata.

Level 1 Metadata is carried in a `DMSegment` containing an `ASPA_Framework` (see 17.1 below), which in turn contains a subclass of `DM_Set` (see 17.2 below). The subclass of `DM_Set` shall be of the class appropriate to the Essence type.

Additionally, the top-level `SourceMob` may contain zero or more `Slots` with `DataDefinition` equal to `DescriptiveMetadata`; containing Level 2 Metadata.

Additionally, the top-level `SourceMob` may contain one `TimelineSlots` with `DataDefinition` equal to `Timecode`; containing `UTCComponents` as defined in section .

Additionally, the top-level `SourceMob` may contain zero or more `Event Slots` with `DataDefinition` equal to `SynchronousDynamicMetadata`; containing `DynamicMarkers` or subclasses as defined in section 18.

Each top-level `SourceMob` shall contain a subclass of `FileDescriptor` appropriate to the Essence type. The top-level `SourceMob` shall contain at least one `Slot` with a `DataDefinition` appropriate to the Essence type. The Segments of such `Slots` may contain a zero-value `SourceReference`, or a `SourceReference` to a lower-level `SourceMob`.

Each lower-level `SourceMob` shall contain an `ImportDescriptor` with a `Locator` naming the file that was imported to create the top-level `SourceMob` and `EssenceData` object. The lower-level `SourceMob` shall contain at least one `Slot` with a `DataDefinition` appropriate to the Essence type. The Segments of such `Slots` shall contain a zero-value `SourceReference` .

6.9 *Slot*

The ASPA Profile does not alter this AAF Class Specification in any way.

6.10 TimelineSlot

The ASPA Profile does not alter this AAF Class Specification in any way.

6.11 EventSlot

The ASPA Profile does not alter this AAF Class Specification in any way.

6.12 StaticSlot

The ASPA Profile does not alter this AAF Class Specification in any way.

6.13 KLVData

The ASPA Profile does not alter this AAF Class Specification in any way.

Note: the SMPTE KLV Sets contained within `KLVData` objects may include any of the MISB-defined KLV Sets including Security Metadata Sets (MISB STD 0102), Predator Standard Metadata Sets (MISB EG-0104) UAS Datalink LDS (MISB STD 0601) and so on.

6.14 TaggedValue

The ASPA Profile does not alter this AAF Class Specification in any way.

6.15 Parameter

The ASPA Profile does not alter this AAF Class Specification in any way.

6.16 ConstantValue

The ASPA Profile does not alter this AAF Class Specification in any way.

6.17 VaryingValue

The ASPA Profile does not alter this AAF Class Specification in any way.

6.18 ControlPoint

The ASPA Profile does not alter this AAF Class Specification in any way.

6.19 Locator

The ASPA Profile does not alter this AAF Class Specification in any way.

6.20 NetworkLocator

The ASPA Profile does not alter this AAF Class Specification in any way.

6.21 TextLocator

The ASPA Profile does not alter this AAF Class Specification in any way.

7 Component Classes

7.1 Component

The ASPA Profile does not alter this AAF Class Specification in any way.

7.2 Transition

The ASPA Profile does not alter this AAF Class Specification in any way.

7.3 Segment

The ASPA Profile does not alter this AAF Class Specification in any way.

7.4 Sequence

The ASPA Profile alters this AAF Class Specification as follows:

ASPA Files shall not contain any `Sequence` objects

7.5 Filler

The ASPA Profile alters this AAF Class Specification as follows:

7.5.1 Numerical Constraints

ASPA Files shall not contain any `Filler` objects.

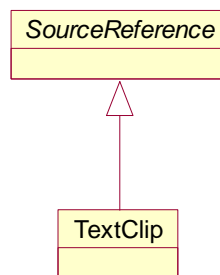
7.6 SourceReference

7.6.1 Extensions

The ASPA Profile defines the `TextClip` subclass of `SourceReference`, as follows:

`TextClip` has a weak reference to a `Slot` describing text essence data.

`TextClip` is an abstract class and is a subclass of `SourceReference`.



The `TextClip` class does not define any properties.

`TextClip` references a `Mob Slot` containing text essence data.

7.7 SourceClip

The ASPA Profile does not alter this AAF Class Specification in any way.

7.8 Event

The ASPA Profile does not alter this AAF Class Specification in any way.

7.9 CommentMarker

The ASPA Profile does not alter this AAF Class Specification in any way.

7.10 DescriptiveMarker

The ASPA Profile alters this AAF Class Specification as follows:.

7.10.1 Extensions

The ASPA Profile defines the `DynamicMarker` subclass of `DescriptiveMarker`, and the `DynamicClip` subclass of `DynamicMarker`, as described in section 18 below.

7.11 GPITrigger

The ASPA Profile alters this AAF Class Specification as follows:

7.11.1 Numerical Constraints

ASPA Files shall not contain any `GPITrigger` objects.

7.12 Timecode

The ASPA Profile does not alter this AAF Class Specification in any way.

7.13 TimecodeStream

The ASPA Profile does not alter this AAF Class Specification in any way.

7.14 TimecodeStream12M

The ASPA Profile does not alter this AAF Class Specification in any way.

7.15 Edgecode

The ASPA Profile alters this AAF Class Specification as follows:

7.15.1 Numerical Constraints

ASPA Files shall not contain any Edgecode objects.

7.16 Pulldown

The ASPA Profile alters this AAF Class Specification as follows:

7.16.1 Numerical Constraints

ASPA Files shall not contain any Pulldown objects.

7.17 OperationGroup

The ASPA Profile alters this AAF Class Specification as follows:

7.17.1 Numerical Constraints

ASPA Files shall not contain any OperationGroup objects.

7.18 NestedScope

The ASPA Profile alters this AAF Class Specification as follows:

7.18.1 Numerical Constraints

ASPA Files shall not contain any NestedScope objects.

7.19 ScopeReference

The ASPA Profile alters this AAF Class Specification as follows:

7.19.1 Numerical Constraints

ASPA Files shall not contain any ScopeReference objects.

7.20 Selector

The ASPA Profile alters this AAF Class Specification as follows:

7.20.1 Numerical Constraints

ASPA Files shall not contain any Selector objects.

7.21 EssenceGroup

The ASPA Profile alters this AAF Class Specification as follows:

7.21.1 Numerical Constraints

ASPA Files shall not contain any EssenceGroup objects.

8 Definition Classes

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

9 Essence Data Classes

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

9.1 *EssenceData*

The ASPA Profile does not alter this AAF Class Specification in any way.

10 Standard Essence Descriptor Classes

10.1 *EssenceDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

10.2 *FileDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

10.3 *DigitalImageDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

10.4 *CDCIDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

10.5 *RGBADescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

10.6 *TapeDescriptor*

The ASPA Profile alters this AAF Class Specification as follows:

10.6.1 Numerical Constraints

ASPA Files shall not contain any *TapeDescriptor* objects.

10.7 *FilmDescriptor*

The ASPA Profile alters this AAF Class Specification as follows:

10.7.1 Numerical Constraints

ASPA Files shall not contain any *FilmDescriptor* objects.

11 Essence Descriptor Classes for Non-Normative Essence Types

11.1 *WAVEDescriptor*

The ASPA Profile alters this AAF Class Specification as follows:

11.1.1 Numerical Constraints

ASPA Files shall not contain any *WaveDescriptor* objects.

11.2 *AIFCDescriptor*

The ASPA Profile alters this AAF Class Specification as follows:

11.2.1 Numerical Constraints

ASPA Files shall not contain any *AIFCDescriptor* objects.

11.3 *TIFFDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

12 Essence Descriptor Classes for Common Compressed Picture Types

12.1 *MPEG2VDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

12.2 *DVDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

12.3 *JFIFDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

13 Essence Descriptor Classes for Sound Essence Types

13.1 *SoundDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

13.2 *PCMDDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

13.3 *PCM8Descriptor*

This section of the AAF Specification is presently not complete.

ASPA Files shall not contain any `PCM8Descriptor` objects.

13.4 *AES3PCMDDescriptor*

The ASPA Profile does not alter this AAF Class Specification in any way.

13.5 *NonPCMDDescriptor*

This section of the AAF Specification is presently not complete.

The ASPA Profile does not alter this AAF Class Specification in any way.

14 Essence Descriptor Classes for Multiple and Generic Container Essence Types

14.1 *MultipleDescriptor*

The ASPA Profile alters this AAF Class Specification as follows:

14.1.1 Numerical Constraints

`MultipleDescriptors` in ASPA Files may contain `RP217Descriptor` objects or `MPEG2MetadataDescriptor` objects.

ASPA Files shall not set the `EssenceContainer` property to `GC_PS` or `GC_PES` or `GC_ES` (defined in the SMPTE Labels Registry).

14.1.2 Semantic Constraints

ASPA Files may contain only MPEG-2 Transport Streams, with or without RP217 KLV Private Data Streams.

14.2 MPEG2SysDescriptor

The ASPA Profile alters this AAF Class Specification as follows:

14.2.1 Numerical Constraints

ASPA Files may contain instances of this class.

14.2.2 Semantic Constraints

ASPA Files may contain only MPEG-2 Transport Streams, with or without RP217 KLV Private Data Streams.

14.3 SysDescriptor

ASPA Files shall not contain any SysDescriptor objects.

14.4 AuxDescriptor

ASPA Files shall not contain any AuxDescriptor objects.

15 Descriptors for Physical Essence

AAF V1.1 defines the following additional Descriptors:

15.1 PhysicalDescriptor

The PhysicalDescriptor class is an abstract superclass which is the parent class for all descriptors of Essence which are indirectly manipulated by AAF applications. It is a peer of the FileDescriptor class (which is the parent class for all descriptors of Essence which are directly manipulated by AAF applications).

The PhysicalDescriptor class is a subclass of the EssenceDescriptor class. It is defined by the AAF Specification V1.1. The ASPA Profile does not alter this AAF Class Specification in any way.

PhysicalDescriptor does not add any new properties to EssenceDescriptor.

15.2 ImportDescriptor

An ImportDescriptor specifies the external file that was imported to create a SourceMob and EssenceData object.

An ImportDescriptor is a concrete subclass of AbstractPhysicalDescriptor. It is defined by the AAF Specification V1.1. The ASPA Profile does not alter this AAF Class Specification in any way.

ImportDescriptor does not define any new properties.

15.3 RecordingDescriptor

ASPA Files shall not contain any RecordingDescriptor objects.

15.4 AuxiliaryFileDescriptor

AuxiliaryFileDescriptor specifies an auxiliary file to be included in an ASPA file. It is defined by the AAF Specification V1.1. The ASPA Profile does not alter this AAF Class Specification in any way.

AuxiliaryFileDescriptor is a concrete subclass of AbstractPhysicalDescriptor.

AuxiliaryFileDescriptor adds the following properties:

Property Name	Type	Explanation
---------------	------	-------------

Property Name	Type	Explanation
MimeType	String	the registered MIME media type used by the data as defined in RFC 2046 and registered according to RFC 2048. Example: L"text/html" Required.
CharSet	String	the registered character set used by the internal and external representation of the data as defined in RFC 2048 and http://www.iana.org/assignments/character-sets Example: L"ISO-8859-1" Optional.

16 Additional Descriptors for ASPA Profile

The ASPA Profile defines the following additional Descriptors:

16.1 *RP217Descriptor*

The `RP217Descriptor` class specifies how KLV packets are contained within an MPEG-2 Systems Stream in a `FileSourceMob` in an ASPA File.

The `RP217Descriptor` class is a subclass of the `DataEssenceDescriptor` class.

`RP217Descriptor` adds the following properties:

Property Name	Type	Explanation
RP217DataStreamPID	UInt16	The ISO 13818-1 Transport Stream PID for the KLV PDS stream Required.
RP217VideoStreamPID	UInt16	The ISO 13818-1 Transport Stream PID for the Video stream Required.

16.1.1 Numerical Constraints

ASPA Files may contain instances of this class.

16.1.2 Semantic Constraints

The `ContainerFormat` property of the `FileDescriptor` shall be set to the constant value for KLVA as defined in the SMPTE Labels Registry: 0x060e2b34 04010102 0D010301 02090602. This corresponds to MPEG-2 TS, PES private data, clip wrapping.

The KLV PDS stream may contain KLV conforming to MISB EG 0104, Std. 0601 and Std 0102.

16.2 *MPEG2MetadataDescriptor*

The `MPEG2MetadataDescriptor` class specifies how metadata packets are contained within an MPEG-2 Systems Stream in a `FileSourceMob` in an ASPA File.

The `MPEG2MetadataDescriptor` class is a subclass of the `FileDescriptor` class.

The `MPEG2MetadataDescriptor` does not add any new properties to the `FileDescriptor`.

16.2.1 Numerical Constraints

ASPA Files may contain instances of this class.

16.2.2 Semantic Constraints

The `ContainerFormat` property of the `FileDescriptor` shall be set to the registered value for KLV formatted per ISO 13818-1:2000- Amd 1 as defined in the SMPTE RP224 Labels Registry:.

16.3 NITFDescriptor

The `NITFDescriptor` class specifies how NITF images are contained within a `FileSourceMob` in an ASPA File.

The `NITFDescriptor` class is a subclass of the `FileDescriptor` class.

The `NITFDescriptor` does not add any new properties to the `FileDescriptor`.

16.3.1 Numerical Constraints

ASPA Files may contain instances of this class.

16.3.2 Semantic Constraints

The `ContainerFormat` property of the `FileDescriptor` shall be set to the constant value for NITF, which shall be registered in the SMPTE RP224 Labels Registry.

16.4 ParsedTextDescriptor

`ParsedTextDescriptor` specifies a text file to be included in an ASPA file.

`ParsedTextDescriptor` is an abstract subclass of `FileDescriptor`.

`ParsedTextDescriptor` adds the following properties:

Property Name	Type	Explanation
Encoding	String	the registered character set used by the external representation of the data as defined in RFC 2048 and http://www.iana.org/assignments/character-sets Example: L"UTF-8" Required.

16.5 SGMLDescriptor

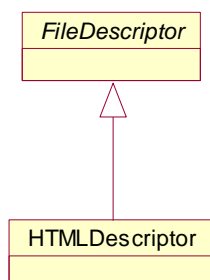
`SGMLDescriptor` is an abstract subclass of `ParsedTextDescriptor`.

The `SGMLDescriptor` does not add any new properties to the `ParsedTextDescriptor`.

16.6 HTMLDescriptor

`HTMLDescriptor` specifies that the essence data is in HTML text format.

`HTMLDescriptor` is a concrete subclass of `SGMLDescriptor`. An `HTMLDescriptor` object is owned by a `File SourceMob` object.



An `HTMLDescriptor` object specifies that the `File SourceMob` describes an HTML object, which contains text, formatted according to the HTML standard.

`HTMLDescriptor` adds the following properties:

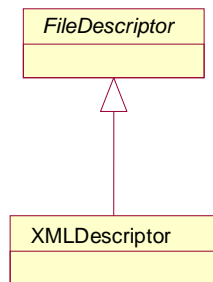
Property Name	Type	Explanation
<code>DocType</code>	String	the complete <code><!DOCTYPE ></code> declaration for this HTML document as defined in the relevant www.w3c.org/TR documents Required.

Example: `L"<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">"`

16.7 XMLDescriptor

`XMLDescriptor` specifies that the essence data is in XML text format.

`XMLDescriptor` is a concrete subclass of `TextFileDescriptor`. An `XMLDescriptor` object is owned by a `File SourceMob` object.



`XMLDescriptor` adds the following properties:

Property Name	Type	Explanation
<code>DefaultNamespaceURI</code>	String	the URI of the default namespace for this XML document as defined in the relevant www.w3c.org/TR documents Example: <code>L"http://www.smppte.org/test"</code> Required
<code>NamespaceTags</code>	StringArray	the Namespace Tags used in QNames in this XML document Example: <code>L"aaf", L"xsi"</code> Optional
<code>NamespaceURIs</code>	StringArray	the URIs associated with Namespace Tags used in QNames in this XML document Example: <code>L"http://www.aafassociation.org/test"</code> , <code>L"http://www.w3.org/2001/XMLSchema-instance"</code> Optional

An `XMLDescriptor` object specifies that the `File SourceMob` describes an XML object, which contains text, formatted according to the XML standard.

16.8 LIDARDescriptor

The LIDARDescriptor class specifies how Lidar files are contained within a FileSourceMob in an ASPA File.

The LIDARDescriptor class is a subclass of the DataEssenceDescriptor class.

The LIDARDescriptor does not add any new properties beyond those of the DataEssenceDescriptor class.

16.8.1 Numerical Constraints

ASPA Files may contain instances of this class.

16.8.2 Semantic Constraints

The ContainerFormat property of the FileDescriptor shall be set to the constant value GC_LAS_V1, which shall be registered in the MISB Metadata Registry.

16.9 LVSDDescriptor

The LVSDDescriptor class specifies how Large Volume Streaming Data (LVSD) essence is contained within a FileSourceMob in an ASPA File.

The LVSDDescriptor class is a subclass of the RGBADescriptor class.

The LIDARDescriptor does not add any new properties beyond those of the DataEssenceDescriptor class.

16.9.1 Numerical Constraints

ASPA Files may contain instances of this class.

16.9.2 Semantic Constraints

LVSD essence shall be composed of sequences of JPEG2000 codestreams (one for each imagery source) in accordance with SMPTE 422M-2006. LVSDDescriptor instances shall include JP2KSubDescriptors that indicate the specific JP2K profile used to compress the images.

ASPA Files describe LVSD essence by default as sequences of images from a single image source (e.g. camera). To accommodate imagery that is interleaved differently (for example, streams that contain several co-timed images from different source), these sequences may be as short as a single image.

The current version of the ASPA spec does not address description or carriage of mosaics of images

The ContainerFormat property of the FileDescriptor shall be set to the constant value GC_LVSD_V1, which shall be registered in the MISB Metadata Registry.

17 Dynamic Metadata

(This chapter of the AAF Specification is presently intentionally unused, reserved for specifications of additional AAF Classes).

SMPTE377M MXF Format and SMPTE EG42 MXF Descriptive Metadata define abstract classes for Descriptive Metadata. The ASPA Profile defines concrete subclasses for Dynamic Metadata, as detailed in the following subsections.

ASPA Files may include as optional properties of Descriptive Metadata classes any other attributes from the D&R IDM which apply to all Product Formats, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a PropertyDefinition for the ClassDefinition of the appropriate class in the MetaDictionary of this ASPA File.

Methodology for properly adding properties is as follows:

Every new ASPA to D&R IDD mapping that is added will generate a new row in the D&R IDM (many rows if a single ASPA property maps to multiple D&R IDM attributes)

When a new property is required to be added to ASPA, the following steps must be carried out:

1. Ensure the property is present in the D&R IDD

(Ensure it is also present in ISO 19115)

2. Identify the appropriate AAF Class

If the Property applies to a single Product_Format, use the ASPA_DM_Set for that Product_Format

If the Property applies to multiple concrete classes, use the ASPA_DM_Set superclass (the AAF superclass which they all have in common)

3. Choose a symbolic name for the Property (it need not be unique beyond the direct ancestor and descendant classes, but it must not contain punctuation other than _)

4. Identify the data type (from AAF Types, as recorded in SMPTE Registry)

5. Obtain a SMPTE UL from the appropriate registry (This is the normative reference.)

Determine which Registry applies (RP210, DoD Public, DoD Private)

RP210: submit entry to SMPTE W25

DoD Public: submit to TBD

DoD Private: submit to TBD

6. Add an entry to the D&R IDM containing:

AAF Class Name

AAF Property Name

AAF Type

SMPTE UL

D&R IDD Entity

D&R IDD Attribute

7. Create an AAF Property Definition in the application code

Each Property registration requires a single API call on the appropriate class Definition

(in future, write and publish the AAF-X schema fragment)

Similar procedures are used to define new Classes

Similar procedures are used to define new Types (including enumerated values)

17.1 ASPA_Framework

The ASPA_Framework class is a container for dynamic metadata defined by ASPA. The classid of the ASPA_Framework class identifies the dynamic metadata as conforming to ASPA.

The `ASPA_Framework` class is a concrete subclass of the `DMFramework` class defined by SMPTE EG42 (aka the `DescriptiveFramework` class defined by AAF V1.1).

`DMFramework` adds the required `SetReference` property:

Property Name	Type	Explanation
<code>SetReference</code>	Strong Reference to <code>DM_Set</code>	The dynamic metadata of the appropriate class. Required.

17.2 *ASPA_DM_Set*

The `ASPA_DM_Set` class is a container for dynamic metadata defined by ASPA.

The `ASPA_DM_Set` class is a abstract subclass of the `DM_Set` class defined by SMPTE EG42 (aka the `DescriptiveObject` class of AAF V1.2)

`DM_Set` predefines the following properties:

Property Name	Type	Explanation
<code>Security_Classification</code>	String Required	The string that represents the Security Classification
<code>Country_Code_Method</code>	String Optional	The coding method used to identify the Non-US classifying country and countries in the releasing instructions. Method is restricted to ISO-3166 two letter, ISO-3166 three letter, FIPS10-4 two letter
<code>Object_Country_Code</code>	String Optional	This maximum 40 character string contains two or three character code(s) as defined by the <code>Country_Code_Method</code> , identifying the country (or countries) that is the object of the video or metadata in the transport stream or file. Multiple codes shall be separated by a semi-colon “;” (no spaces). Multiple codes shall be concatenated in one object country code metadata element entry and shall not be encoded as individual metadata elements.
<code>Non_US_Classification_Country</code>	String Optional	This metadata element contains a value for the Non-US classifying country code.
<code>Caveats</code>	String Optional	All pertinent caveats/codewords from each category of the CAPCO register
<code>Release_Instructions</code>	String Optional	Valid list of country codes to which countries the file is authorized for release. When multiple countries are listed, countries are separated by a space with the originating country listed first.
<code>Classification_Comment</code>	String Optional	Comments pertaining to security
<code>Product_Format</code>	String Required	The code that represents an NSGI standard format for a DATASET (per the NERS Appendix D Table 2), or a format that is available from a Library as an alteration (also known as an export format). The native format in which the NSGI Library stores the data.
<code>Product_Title</code>	String Optional	The name by which the DATASET is known.
<code>Creation_Time</code>	Timestamp Optional	Identifies the date or the date and time that the product was created or last modified.
<code>Classification_Reason</code>	String Optional	This free-text metadata element contains the reason for classification or the citation.

Derived_From	String Optional	This free-text metadata element contains information about the original source file or document from which the classification was derived.
Classified By	String Optional	This free text metadata element identifies the name and type of authority used to classify the data and can contain either the original classification authority name and position or personal identifier.
SCI_Handling Instructions (SHI)	String Optional	This formatted text metadata element contains special handling instructions if material in the transport stream is Top Secret, Secret, or Confidential. ISCI/SHI digraphs, trigraphs, or . compartment names must be added identifying a single or a combination of special handling instructions. A single entry shall be ended with a double forward slash "//". Multiple digraphs, trigraphs, or compartment names shall be separated by a single forward slash "/" and the last entry shall be ended with a double forward slash "//". Multiple SCI/SHI digraphs, trigraphs, or compartment names shall be concatenated in one metadata element free-text entry and shall not be encoded as individual metadata elements in the Sets.
Declassification Date/Exemption	String Optional	This metadata element provides either a date when the classified material may be automatically declassified or if it is subject to Manual Review (MR) and is exempt from automatic declassification. The declassification date format shall be YYYYMMDD or the letters "MR" shall be used.
Property Name	Type	Explanation

Originators_Name	String Optional	The text that represents the originator.
Originating_Station_ID	String Optional	The identifier that represents the originating organization, system, station or product.

All these properties are defined by the D&R IDM (Rev E). Additional properties may be defined by later revisions of the D&R IDM.

Instances of DM_Set shall include all properties marked as Required, and may include any of the properties marked as Optional.

In addition, instances of DM_Set may include as optional properties any other attributes from the D&R IDM which apply to all Product Formats, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a PropertyDefinition for the ClassDefinition of DM_Set in the MetaDictionary of this ASPA File.

17.3 DM_Set_File

The DM_Set_File class is a container for ASPA level 0 metadata, which is metadata that pertains to the total file. The DM_Set_File class is a concrete subclass of ASPA_DM_Set.

DM_Set_File inherits all properties of ASPA_DM_Set, and predefines no additional properties. Additional properties may be defined by later revisions of the D&R IDM.

Instances of DM_Set_File shall include all properties marked as Required, and may include any of the properties marked as Optional.

The required property Product_Format of the superclass DM_Set shall have the value "AAF_ASPA".

In addition, instances of DM_Set_File may include as optional properties any other attributes from the D&R IDM which apply to the overall ASPA file, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a PropertyDefinition for the ClassDefinition of this class in the MetaDictionary of this ASPA File.

Additionally, the MobID of the MasterMob to which this set is attached is mapped to the D&R IDM DATASET: DATASET Identification Text attribute.

17.4 DM_Set_MPEGKLV_Layer

The DM_Set_MPEGKLV_Layer class is a container for ASPA level 1 metadata, which is metadata that pertains to a product of type MPEGKLV within the file. The DM_Set_MPEGKLV_Layer class is a concrete subclass of ASPA_DM_Set.

DM_Set_MPEGKLV_Layer adds the following properties:

Property Name	Type	Explanation
Image_Source_Device	String Optional	A free text identification of the particular image sensor type and serial number.

Property Name	Type	Explanation
Start_Date_Time	Timestamp Optional	The date and time an image was collected.
Bounding_Rectangle	GeographicArea Optional	Defines the boundary for an area of inclusion or exclusion for an IMAGE.
Platform_Designation	String Optional	Platform ID. From KLV Platform Designation.
Target_ID	String Optional	Combination of BE Number, OSUFFIX, and Country Code. From KLV Target Id.
Duration	UInt64 Optional	Duration of the MPEG essence, in milliseconds.
Motion_Imagery_ID	String Optional	Multi-field identifier derived from the MI stream using the algorithm defined in RP 0608.1 The ID is used to identify each unique clip.

Instances of `DM_Set_MPEGKLV_Layer` shall include all properties marked as Required, and may include any of the properties marked as Optional.

The required property `Product_Format` shall have the value "MPEGKLV".

All these properties are defined by the D&R IDM (Rev E). Additional properties may be defined by later revisions of the D&R IDM.

In addition, instances of `DM_Set_MPEGKLV_Layer` may include as optional properties any other attributes from the D&R IDM which apply to this Product Format, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a `PropertyDefinition` for the `ClassDefinition` of this class in the `MetaDictionary` of this ASPA File.

Additionally, the `MobID` of the `SourceMob` to which this set is attached is mapped to the D&R IDM DATASET: DATASET Identification Text attribute.

17.5 DM_Set_NITF_Layer

The `DM_Set_NITF_Layer` class is a container for ASPA level 1 metadata, which is metadata that pertains to a product of type NITF within the file. The `DM_Set_NITF_Layer` class is an abstract subclass of `ASPA_DM_Set`.

ASPA Files may contain one of the concrete subclasses of `DM_Set_NITF_Layer`:

- `DM_Set_NITF21_Layer`
- `DM_Set_NITF20_Layer`
- `DM_Set_NSIF10_Layer`

`DM_Set_NITF_Layer` adds the following properties:

Property Name	Type	Explanation
Date_and_Time	AAFTimestamp Optional	This field shall contain the time of the image acquisition. From NITF IDATIM.

Property Name	Type	Explanation
Target_ID	String Optional	For NITF 2.0: Combination of BE Number, Functional Category Code and Country Code. For NITF 2.1 and NSIF 1.0: Combination of BE Number, OSUFFIX, and Country Code from NITF TGTID.
Geographic_Location	Geographic Polygon Optional	Defines the boundary for an area of inclusion or exclusion for an IMAGE. This shall be calculated as the bounding rectangle that includes all IMAGE layers within the NITF file.

All these properties are mapped into the D&R IDM (Rev E).

Instances of `DM_Set_NITF_Layer` shall include all properties marked as Required, and may include any of the properties marked as Optional.

The required property `Product_Format` shall have the value "NITF", optionally suffixed by the NITF version number, for example: "NITF02.10".

All these properties are defined by the D&R IDM (Rev E). Additional properties may be defined by later revisions of the D&R IDM.

In addition, instances of `DM_Set_NITF_Layer` may include as optional properties any other attributes from the D&R IDM which apply to this Product Format, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a `PropertyDefinition` for the `ClassDefinition` of this class in the `MetaDictionary` of this ASPA File.

Additionally, the `MobID` of the `SourceMob` to which this set is attached is mapped to the D&R IDM DATASET: DATASET Identification Text attribute.

17.6 DM_Set_JFIF_Layer

The `DM_Set_JFIF_Layer` class is a container for ASPA level 1 metadata, which is metadata that pertains to a product of type JFIF within the file. The `DM_Set_JFIF_Layer` class is a concrete subclass of `ASPA_DM_Set`.

`DM_Set_JFIF_Layer` adds the following properties:

Property Name	Type	Explanation
Description	String Optional	The text that describes source material for an IMAGE.

All these properties are mapped into the D&R IDM (Rev E)

Instances of `DM_Set_JFIF_Layer` shall include all properties marked as Required, and may include any of the properties marked as Optional.

The required property `Product_Format` shall have the value "JFIF".

All these properties are defined by the D&R IDM (Rev E). Additional properties may be defined by later revisions of the D&R IDM.

In addition, instances of `DM_Set_JFIF_Layer` may include as optional properties any other attributes from the D&R IDM which apply to this Product Format, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a `PropertyDefinition` for the `ClassDefinition` of this class in the `MetaDictionary` of this ASPA File.

Additionally, the `MobID` of the `SourceMob` to which this set is attached is mapped to the D&R IDM DATASET: DATASET Identification Text attribute.

17.7 DM_Set_HTML_Layer

The `DM_Set_HTML_Layer` class is a container for ASPA level 1 metadata, which is metadata that pertains to a product of type HTML within the file. The `DM_Set_HTML_Layer` class is a concrete subclass of `ASPA_DM_Set`.

`DM_Set_HTML_Layer` adds the following properties:

Property Name	Type	Explanation
Description	String Optional	The text that describes source material for an HTML document.
Lang	String Optional	Code indicating the language used on an item.

All these properties are mapped into the D&R IDM (Rev E)

Instances of `DM_Set_HTML_Layer` shall include all properties marked as Required, and may include any of the properties marked as Optional.

The required property `Product_Format` shall have the value "HTML".

All these properties are defined by the D&R IDM (Rev E). Additional properties may be defined by later revisions of the D&R IDM.

In addition, instances of `DM_Set_HTML_Layer` may include as optional properties any other attributes from the D&R IDM which apply to this Product Format, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a `PropertyDefinition` for the `ClassDefinition` of this class in the `MetaDictionary` of this ASPA File.

Additionally, the `MobID` of the `SourceMob` to which this set is attached is mapped to the D&R IDM DATASET: DATASET Identification Text attribute.

17.8 DM_Set_LIDAR_Layer

The `DM_Set_LIDAR_Layer` class is a container for ASPA level 1 metadata, which is metadata that pertains to a product of type LIDAR within the file. The `DM_Set_LIDAR_Layer` is a subclass of `ASPA_DM_Set`.

The ASPRS LIDAR standard (LAS) file format, as the industry standard open file format, is currently the only supported encapsulation of LIDAR data.

`DM_Set_LIDAR_Layer` contains the following properties:

Property Name	Type	Explanation
Project_ID_Data	UID Optional	Global Unique Identifier intended to provide space for uniquely identifying a LIDAR project. Derived from LASF 1.1 PUBLIC HEADER fields GUID data 1 through 4.
File_Source_ID	Uint32 Optional	A LIDAR project can indicate a number of unique sources, such as files containing an original flight line or the result of a merge or extract operation. This field uniquely identifies these sources. Derived from LASF 1.1 PUBLIC HEADER field File Source ID.
LIDAR_Bounding_Rectangle	GeographicArea Optional	Defines the boundary geographic extents of coverage for the LIDAR essence being described. Computed from Georeferencing variable length records, if present, in LASF 1.1.
System_Identifier	String Optional	Specifies the system which gathered the LIDAR data. Derived from LASF 1.1 PUBLIC HEADER field System Identifier.

Instances of `DM_Set_LIDAR_Layer` shall include all properties marked “Required”, and may include any of the properties marked as “Optional”.

The required property `Product_Format` shall have the value “LASF”, suffixed by the LASF version number. Possible values are “LASF1.0” and “LASF1.1”.

In addition, instances of `DM_Set_LIDAR_Layer` may include as optional properties any other attributes from the D&R IDM which apply to this Product Format, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a `PropertyDefinition` for the `ClassDefinition` of this class in the `MetaDictionary` of this ASPA File.

17.9 DM_Set_LVSD_Layer

The `DM_Set_LVSD_Layer` class is a container for ASPA level 1 metadata, which is metadata that pertains to a product of type LVSD within the file. The `DM_Set_LVSD_Layer` is a subclass of `ASPA_DM_Set`.

`DM_Set_LVSD_Layer` contains the following properties:

Property Name	Type	Explanation
---------------	------	-------------

LVSD_Bounding_Rectangle	GeographicArea Optional	Defines the boundary geographic extents of coverage for the LVSD essence being described.
Computed from the GeographicQuadrilateralStream data in the SynchronizedDynamicMetadata slot of the FilePackage .		

Instances of `DM_Set_LVSD_Layer` shall include all properties marked “Required”, and may include any of the properties marked as “Optional”.

The required property `Product_Format` (inherited from `ASPA_DM_Set`) shall have the value “LVSD”

In addition, instances of `DM_Set_LVSD_Layer` may include as optional properties any other attributes from the D&R IDM which apply to this Product Format, provided that the D&R IDM defines a SMPTE Universal Label for that attribute. Methodology for adding these properties is described in above. The SMPTE Universal Label shall be used as the unique identifier of this attribute in a `PropertyDefinition` for the `ClassDefinition` of this class in the `MetaDictionary` of this ASPA File.

18 DynamicMarker Classes

(This chapter of the AAF Specification is presently intentionally unused, reserved for specifications of additional AAF Classes).

18.1 DynamicMarker Class

The `DynamicMarker` class is a container for synchronous dynamic metadata defined by ASPA.

The `DynamicMarker` class is a concrete subclass of the `DescriptiveMarker` class of AAF V1.2(aka the `DMSegment` class defined by SMPTE EG42).

The reference time for the synchronous dynamic metadata is carried in the `Position` property of the `Event` superclass. The synchronous dynamic metadata itself is carried in the `KLVDData` property of the `Component` superclass.

The `DynamicMarker` class adds the following properties to `DescriptiveMarker`:

Property Name	Type	Explanation
ToleranceMode	<code>ToleranceModeType</code>	An integer that enumerates the mode of determining the reference time of this <code>DynamicMarker</code> . Allowed values are as follows: Estimated Assumed Precise Window Interpolated The meaning of these modes is described in section 21.4 below Required.

Property Name	Type	Explanation
InterpolationMethod	WeakReference InterpolationDefinition	A reference to the well-known interpolation method used to interpolate metadata values to the reference time. Optional
ToleranceWindow	Indirect	The time window associated with the ToleranceMode, if any. If positive, the window shall be centered on the given reference time. If negative, the window shall end at the given reference time. Optional. This is an Indirect type – the value starts with the 16 byte identifier of the actual type.

Note: if the actual type of the ToleranceWindow is Length, the size of the window shall be calculated using the edit rate of the MobSlot in which the DynamicMarker is contained. This is to match the semantics of the SourceClip class. In all other cases, the window shall be calculated in absolute terms

Note: the ASPA specification 1.00 does not provide any standard method to indicate the estimated error in a data value.

18.2 DynamicClip Class

The DynamicClip class contains a reference to the source of synchronous dynamic metadata defined by ASPA.

The DynamicClip class is a concrete subclass of DynamicMarker.

A DynamicClip may be used in place of a DynamicMarker to indicate the SourceMob, slot(s) and position from which the synchronous dynamic metadata value is obtained. If the KLVDData property of the Component superclass is not present, the value must be obtained from the indicated source whenever it is required. Conversely, if the KLVDData property is present, it shall contain a copy of the referenced synchronous dynamic metadata.

The DynamicClip class adds the following properties to DynamicMarker:

Property Name	Type	Explanation
SourceMobID	MobIDType	The MobID of the SourceMob from which the synchronous dynamic metadata is obtained. Optional. A distinguished value of 0 indicates that the source of the metadata is unknown. If this property is not present, the source slot refers to a track in the same Mob.
SourceSlotIDs	UInt32Array	The SlotIDs of the slot or slots in the SourceMob from which the synchronous dynamic metadata is obtained. Optional
SourceIndex	Indirect	The index of the dynamic metadata within the referenced source, using the type given. Optional. This is an Indirect type – the value starts with the 16 byte identifier of the actual type. Normally, this value will be the identifier of the Position type

Property Name	Type	Explanation
SourceSpecies	Indirect	The selectors of the elements from the source that are used in the referring MobSlot. All other elements from the SourceMob shall be ignored. Optional. This is an Indirect type – the value starts with the 16 byte identifier of the actual type. Normally, this value will be the identifier of the “ArrayOfAUID” type.

Notes: if the actual type of the SourceIndex is Position, the position in the SourceMob shall be calculated using the edit rate of the MobSlot in which the DynamicClip is contained. This is to match the semantics of the SourceClip class. In all other cases, the SourceIndex shall be calculated in the frame of reference of the SourceMob.

19 Support Classes for ASPA

(This chapter of the AAF Specification is presently intentionally unused, reserved for specifications of additional AAF Classes).

19.1 Geographic Area

The Class Geographic Area has the following properties:

Property Name	Type	Explanation
GeographicArea_NorthWest	Geographic_Coordinate	The NorthWest corner point of the area
GeographicArea_SouthEast	Geographic_Coordinate	The SouthEast corner point of the area
GeographicArea_SourceDatum	String Optional	Code indicating the source datum from which the coordinates are measured, per DIGEST spec. Default value = “WGE” Other values: “NAR”, “NAS”

19.2 Geographic Polygon

The Class Geographic Polygon has the following properties:

Property Name	Type	Explanation
GeographicPolygon_Coords	Geographic_Coordinate_Array	The corner points of the polygon, in clockwise sequence
GeographicPolygon_SourceDatum	String Optional	Code indicating the source datum from which the coordinates are measured, per DIGEST spec Default value = “WGE” Other values: “NAR”, “NAS”

19.3 Geographic Quadrilateral Stream Class

The Class GeographicQuadrilateralStream is used to gather the position information for a single image or a sequence of images into a data structure that may be encoded as a single KLV packet.

The Class GeographicQuadrilateralStream contains a GeographicQuadrilateralStream_SourceDatum and a variable-length array of GeographicQuadrilaterals, all of which are measured from the same datum.

The Class `GeographicQuadrilateralStream` has the following properties:

Property Name	Type	Explanation
<code>GeographicQuadrilateralStream_Quadrilaterals</code>	<code>Geographic_QuadrilateralArray</code>	The area covered by each image in the sequence
<code>GeographicQuadrilateralStream_SourceDatum</code>	<code>StringOptional</code>	Code indicating the source datum from which the coordinates are measured, per DIGEST spec Default value = "WGE" Other values: "NAR", "NAS"

19.4 UTCComponent Class

The Class `UTCComponent` is used to describe an interval of clock time. `UTCComponent` is used in place of `Timecode` for synchronization of motion imagery and dynamic metadata that is not tied intimately to video recording frame rates.

`UTCComponent` has the following properties:

Property Name	Type	Explanation
<code>StartUTC</code>	<code>UTCString</code>	The clock time of the start of the time interval

20 Unused Chapter

This chapter of the AAF Specification is presently intentionally unused.

21 Data Types

The ASPA Profile adds the following Data Types to the AAF Specification:

21.1 Fix32Dec3

The Type `Fix32Dec3` is used to represent a value with 3 decimal places. In ASPA files, geographic Latitude and Longitude are measured in 1/1000 of an arc-second and are represented as `Fix32Dec3` values.

21.2 Geographic Coordinate

The Type `Geographic Coordinate` is a Record with two members: Latitude and Longitude, both of Type `Fix32Dec3`.

21.3 Geographic Coordinate Array

The Type `Geographic Coordinate Array` contains a variable-length array of `Geographic Coordinates` as used in a `Geographic Polygon`

21.4 ToleranceModeType

The type `ToleranceModeType` enumerates the mode of determining the reference time of a `DynamicMarker`. Allowed values are as follows:

Symbol	Value	Explanation
<code>Estimated</code>	0	The value at the given reference time is estimated, not using any known interpolation method.

Assumed	1	The data was observed and the time of observation is assumed to be as given. No analytical weight can be given to the observation time or the size of the Window - they are guesstimates. Any interpolation is suspect
Precise	2	The data was observed at the precise reference time given.
Window	3	The data was observed sometime within a window of time relative to the given reference time.
Interpolated	4	The data value is the interpolated value that would be expected at the given reference time, using the given InterpolationMethod over the actual data received in the given time Window relative to the given reference time.

21.5 Geographic Quadrilateral

The Type `Geographic Quadrilateral` contains an array of four `Geographic_Coordinates`, in the sequence Top Left, Top Right, Bottom Right, Bottom Left in sensor field of view, as used in a `Geographic Quadrilateral Array`.

21.6 Geographic Quadrilateral Array

The Type `Geographic Quadrilateral Array` contains a variable-length array of `Geographic Quadrilaterals`, as used in a `Geographic Quadrilateral Stream`

21.7 UTCString

The Type `UTCString` is a derived type based on `String`, contains a single GPS timestamp, formatted according to ISO 8601 with timezone specifier and a time resolution of 1 millisecond or better, as used in a `UTCComponent` class.

22 DataDefinitions

Note that the approved Data Definitions are changed in V1.1 compared with AAF V1.0.1.

The ASPA Profile extends the provisions of this chapter of the AAF Specification as follows:

22.1 DynamicMetadata

ASPA defines the label `DynamicMetadata`, which is registered in the SMPTE Labels Registry RP224 v7

22.2 SynchronousDynamicMetadata

ASPA defines the label `SynchronousDynamicMetadata`, which shall be registered in the MISB Registry

23 Extensible Enumerations

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

24 Operation Groups

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

25 Tutorial on Compositions

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

26 Tutorial on Describing Essence

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

27 MetaDefinitions

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

28 Extensions

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

29 Bibliography

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

30 Conventions

The ASPA Profile does not alter the provisions of this chapter of the AAF Specification in any way.

Annex A Previous Versions

This annex lists substantive changes from previous versions of the ASPA Profile.

Changes in Version 1.5

- 2 References updated.
- 3 This document supports the ASPA Browser version 5.0 that uses AAF SDK version 1.1.4 and mxflib version 1.2.
- 6.13 Support for Standard 0601.4 and Standard 0102.8 included.

Editorial Clarifications in Version V1.4

- 1 added statement of current practice and intent

Editorial Clarifications in Version V1.3a

- 6.8 clarified that events may be `DynamicMarkers` or subclasses (such as `DynamicClip`)
- 17.8 clarified name of `LIDAR_Bounding_Rectangle`
- 17.9 clarified name of `LVSD_Bounding_Rectangle`
- 18.2 clarified semantics of `DynamicClip` to match AAF spec

Changes from Version 1.2

- 6.8 added provision for `SynchronousDynamicMetadata` slots
- 6.8 added provision for `Timecode` slots
- 16.9 added `LVSDDescriptor`
- 17.9 added `DM_Set_LVSD_Layer`
- 19.3 added `GeographicQuadrilateralStream` class
- 19.4 added `UTCComponent` class
- 20.3 added `SynchronousDynamicMetadata` label
- 21.5 added `GeographicQuadrilateral` type with coordinates in clockwise sequence starting top left
- 21.6 added `GeographicQuadrilateralArray` type
- 21.7 added `UTCString` type

Changes from Version 1.0

Added `DM_Set_LAS_Layer` and `LidarDescriptor`.

Added reference to MISB EG0602.

Added `Duration` property to `DM_Set_MPEGKLV_Layer`

Clarified `SourceDatum` property of `Geographic Polygon` and `Rectangle` and corrected examples to match DIGEST specification.

Added `Object_Country_Code` property to `ASPA_DM_Set` layer.

Added `Motion_Imagery_ID` property to `DM_Set_MPEGKLV_layer`.

Changes from Version 0.9

ASPA Profile V0.9 document created December 2004 with ASPA Browser version 2.5 RC1.

Changes from Version 0.8.2

ASPA Profile V0.8.2 is the document distributed August 21st, 2004

Updated references to AAF V1.1 and MXF: SMPTE 377M-2004.

6.5 to 6.9: change “Package” to “Mob” per AAF V1.1

6.8 described SourceMobs for external essence

7.10 inserted DescriptiveMarker

Reordered and renumber chapter 15 to correspond to AAF Specification V1.1 chapter 15. Inserted chapter 16, renumbered chapter 17 and 18, renumbered chapter 17 to 19.

16.1 RP217Descriptor: updated superclass name, updated properties,, updated Container label

16.5 SGMLDescriptor, 16.6 HTMLDescriptor: added intermediate abstract SGMLDescriptor superclass.

Added chapter 18 on DynamicMarkers and DynamicClips

Added section 21.4 on ToleranceMode type

22. Noted that AAF V1.1 changes DataDefinition values.

Changes from Version 0.8

ASPA Profile V0.8 corresponds to the document distributed up to April 8 2004.

The version numbers were updated for the following references: MISP V 2.3, MISB RP 102.2, MISB RP 104.3.

Country_Code_Method is constrained to have one of the following values: ISO-3166 Three Letter, ISO-3166 Two Letter, FIPS 10-4 Two Letter

Release Instructions now specify that Countries are separated by a space when multiple countries are listed.

NonUS_Classification-Country is now spelled Non_US_Classification_Country

Coordinate_System was removed from DM_Set_MPEGKLV_Layer and DM_Set_NITF_Layer. This element is now incorporated into GeographicArea and GeographicPolygon.

Remove Map_Datum_Used from DM_Set_MPEGKLV_Layer. This is in GeographicArea and GeographicPolygon.

Fix class and type names for section 17.1 & 17.2

Changes from Version 0.7

ASPA Profile V0.7 corresponds to the AAF-in-IPL prototype distributed up to February 2004.

In V0.7, ASPA-specific Classes, Properties and Types were identified with temporary UUIDs. In V0.8, all identifications change to SMPTE ULs, and are registered in the appropriate registries:

SMPTE Registry – RP210

SMPTE Registry – RP224

AAF Registry – AAF Association SDK V1.0.2 “AAFMetaDict.xls”

ASPA – “libaspaIDs.h”

In V0.7, ASPA files extended AAF by adding a property to Identification, named KLVDData:

Property Name	Type	Explanation
KLVDData	strong reference to KLVDData	The KLVDData shall include a Security Metadata Set (MISB RP-0102) which specifies the security metadata applicable to the file at the time the file was modified (as recorded in the ModificationTime property of the Identification. Required.

In V0.7, ASPA files carried Level 0 Metadata in the KLVDData property of the Identification object.

In V0.7, ASPA files did not contain any lower-level Source Mobs (section 6.8)

In V0.7, ASPA Files carried DM_Set_MPEGKLVLayer::Bounding_Rectangle as a property of type String.

In V0.7, ASPA Files carried DM_Set_NITFLayer:: Geographic_Location as a property of type String.

In V0.7, level 1 Dynamic Metadata is derived only from the first top-level file Source Mob, and only from the first image layer within an NITF file.